

VIS Übungblatt 1

Backup

Aufgabe 1 (Pflicht, 3 P.)

Nennen Sie drei typische Probleme, die für eine Datensicherung sprechen.

Typische Probleme sind u.a.

1. Hardwaredefekte, also z.B. Ausfall einer Festplatte
2. Diebstahl von Datenträgern wie Festplatten
3. Höhere Gewalt, also z.B. Feuer, Erdbeben etc.

Aufgabe 2 (Optional)

Welche Einflussfaktoren sind bei der Wahl von Backupstrategie und Backup- medium zu berücksichtigen?

Aufgabe 3 (Optional)

Auf welche Punkte sollte bei der Wahl des Backup-Aufbewahrungsortes geachtet werden?

Aufgabe 4 (Pflicht, 4 P.)

Was ist der Unterschied zwischen einem inkrementellen und einem differenziellen Backup? Welche Vor- und Nachteile haben diese Varianten?

Das inkrementelle Backup sichert alle Daten, die nach dem letzten Backup (Voll oder inkrementell) hinzugekommen sind. Ein Vorteil ist hierbei die im Vergleich zu dem differenziellen Backup bessere Speichereffizienz, da nur neue Daten gesichert werden. Zum Wiederherstellen eines vorherigen Standes müssen allerdings das Voll-Backup plus alle bisherigen inkrementellen Backups nacheinander ausgeführt werden, dies bedeutet ggf. mehr Aufwand. Ist zudem ein inkrementelles Backup vor dem aktuellen Backup defekt/gelöscht, kann nur der Stand bis vor diesem wiederhergestellt werden.

Beim differenziellen Backup werden alle Daten, die nach dem letzten Voll-Backup hinzugekommen sind gesichert. Dies kann zu höherem Speicherbedarf führen, vereinfacht aber die Wiederherstellung. Zudem haben differenzielle Backup als Abhängigkeit einzig das letzte Voll-Backup.

RAID

Aufgabe 5 (Pflicht, 1 P.)

Der Einsatz von RAID-Systemen erspart im Regelfall nicht ein Backup. Warum?

RAID-Systeme können die Datensicherung verstärken und ggf. die Verfügbarkeit verbessern, allerdings schützen sie nicht vor versehentlichem bzw. gewollten Löschen von Daten, Viren und möglichen Softwarefehlern.

Aufgabe 6 (Optional)

Sie suchen nach einer fehlertoleranten Speicherlösung. Der Ausfall einer Festplatte soll toleriert werden. In die engere Auswahl kommen RAID 1 und RAID 5. Vergleichen Sie RAID 1 und RAID 5 hinsichtlich Fehlertoleranz, Performance und Kosten.

Aufgabe 7 (Pflicht, 3 P.)

Sie verfügen über 4 Festplatten mit jeweils 500 GB Speicherplatz. Geben Sie für jede der folgenden RAID-Varianten an, wieviel Speicherplatz Sie für Daten nutzen können: a) RAID 0, b) RAID 1, c) RAID 4, d) RAID 5, e) RAID 6 und f) RAID 10.

1. RAID 0
Ein RAID 0-System verteilt die Daten auf den 4 Festplatten (ohne Redundanz). Hier sind also $4 \cdot 500\text{GB} = \mathbf{2\text{TB}}$ Speicherplatz nutzbar.
2. RAID 1
Werden die Daten im RAID 1 nur einmal gespiegelt, wären $2 \cdot 500\text{GB} = \mathbf{1\text{TB}}$ Speicherplatz nutzbar. (Bei höherer Spiegelungszahl entsprechend weniger)
3. RAID 4
Für RAID 4 wird eine Festplatte für die Paritätsinformationen gebraucht, also sind $3 \cdot 500\text{GB} = \mathbf{1.5\text{TB}}$ Speicherplatz nutzbar.
4. RAID 5
Ähnlich wie zu RAID 4, die Paritätsinformationen sind hier zwar verteilt, aber genauso groß, also sind effektiv $3 \cdot 500\text{GB} = \mathbf{1.5\text{TB}}$ Speicherplatz nutzbar.
5. RAID 6
Durch die *Kreuzparität* werden hier zwei Festplatten für die Paritätsinformationen gebraucht, d.h. es sind $2 \cdot 500\text{GB} = \mathbf{1\text{TB}}$ Speicherplatz nutzbar.
6. RAID 10
Bei der verteilte Spiegelung werden 1 Festplatte für die Spiegelung und 2 für die verteilte Spiegelung genutzt, daher ist nur $\mathbf{500\text{GB}}$ Speicherplatz nutzbar.

Aufgabe 8

Gegeben ist folgendes RAID-4-System, bestehend aus drei Datenplatten (HDD-1 bis HDD-3) und einer Paritätsplatte (HDD-P):

- Inhalt HDD-1 01101011
- Inhalt HDD-2 10010011
- Inhalt HDD-3 00011011
- Inhalt HDD-P 11100011

Aufgabe 8a (Optional)

Ist das RAID-4-System in einem konsistenten Zustand?

Aufgabe 8b (Pflicht, 4 P.)

Der Inhalt von HDD-1 soll mit 10010111 überschrieben werden. Berechnen Sie die neue Parität mit und ohne Small-Write Algorithmus.

Ohne Small-Write:

$$\begin{aligned} \text{HDD-1}_{\text{new}} \oplus \text{HDD-2} \oplus \text{HDD-3} &= \text{HDD-P}_{\text{new}} \\ 10010111 \oplus 10010011 \oplus 00011011 &= 00011111 \end{aligned}$$

Mit Small-Write:

$$\begin{aligned} \text{HDD-1}_{\text{new}} \oplus \text{HDD-1}_{\text{old}} \oplus \text{HDD-P}_{\text{old}} &= \text{HDD-P}_{\text{new}} \\ 10010111 \oplus 01101011 \oplus 11100011 &= 00011111 \end{aligned}$$

Aufgabe 8c (Pflicht 6 P.)

Das Codewort im Bild lautet: Plattendiversität.

Ich habe ein einfaches Programm in Java geschrieben, welches nicht sehr dynamisch ist und keine Fehlerbehandlung durchführt, aber das Problem löst:

```
1 public class Main {
2     private final static String disksFolder = ".../VIS/Übung/Blatt01/raid4/";
3     private final static String disk0 = "disk0";
4     private final static String disk2 = "disk2";
5     private final static String diskP = "disk3";
6     private final static int stripSize = 4096;
7
8     public static void main(String[] args) throws IOException {
9         final Path disk0f = Paths.get(disksFolder + disk0);
10        final Path disk2f = Paths.get(disksFolder + disk2);
11        final Path diskPf = Paths.get(disksFolder + diskP);
12
13        byte[] disk0Data = Files.readAllBytes(disk0f);
14        byte[] disk2Data = Files.readAllBytes(disk2f);
15        byte[] diskPData = Files.readAllBytes(diskPf);
16
17        Path data = Paths.get(disksFolder + "data");
18        Files.write(data, readRaidDisks(disk0Data, restoreDisk(disk0Data, disk2Data, diskPData),
19                                     disk2Data));
19    }
20
21    private static byte[] restoreDisk(final byte[] disk0Data, final byte[] disk2Data, final byte[]
22                                     diskPData) {
23        byte[] disk1Data = new byte[diskPData.length];
24
25        //restoring disk with xor composition
26        int i = 0;
27        for (byte diskPDataB : diskPData) {
28            disk1Data[i] = (byte) ((diskPDataB ^ disk0Data[i]) ^ disk2Data[i]);
29            i++;
30        }
31        return disk1Data;
32    }
33
34    private static byte[] readRaidDisks(final byte[] disk0Data, final byte[] disk1Data, final
35                                       byte[] disk2Data) throws IOException {
36        final ByteArrayOutputStream outputStream = new ByteArrayOutputStream();
37
38        //concatenate data strips to get an image of cute kittens
39        for(int i = 0; i < disk0Data.length / stripSize; i++) {
40            int from = stripSize * i;
41            int to = from + stripSize;
42            outputStream.write(Arrays.copyOfRange(disk0Data, from, to));
43            outputStream.write(Arrays.copyOfRange(disk1Data, from, to));
44            outputStream.write(Arrays.copyOfRange(disk2Data, from, to));
45        }
46        return outputStream.toByteArray();
47    }
```